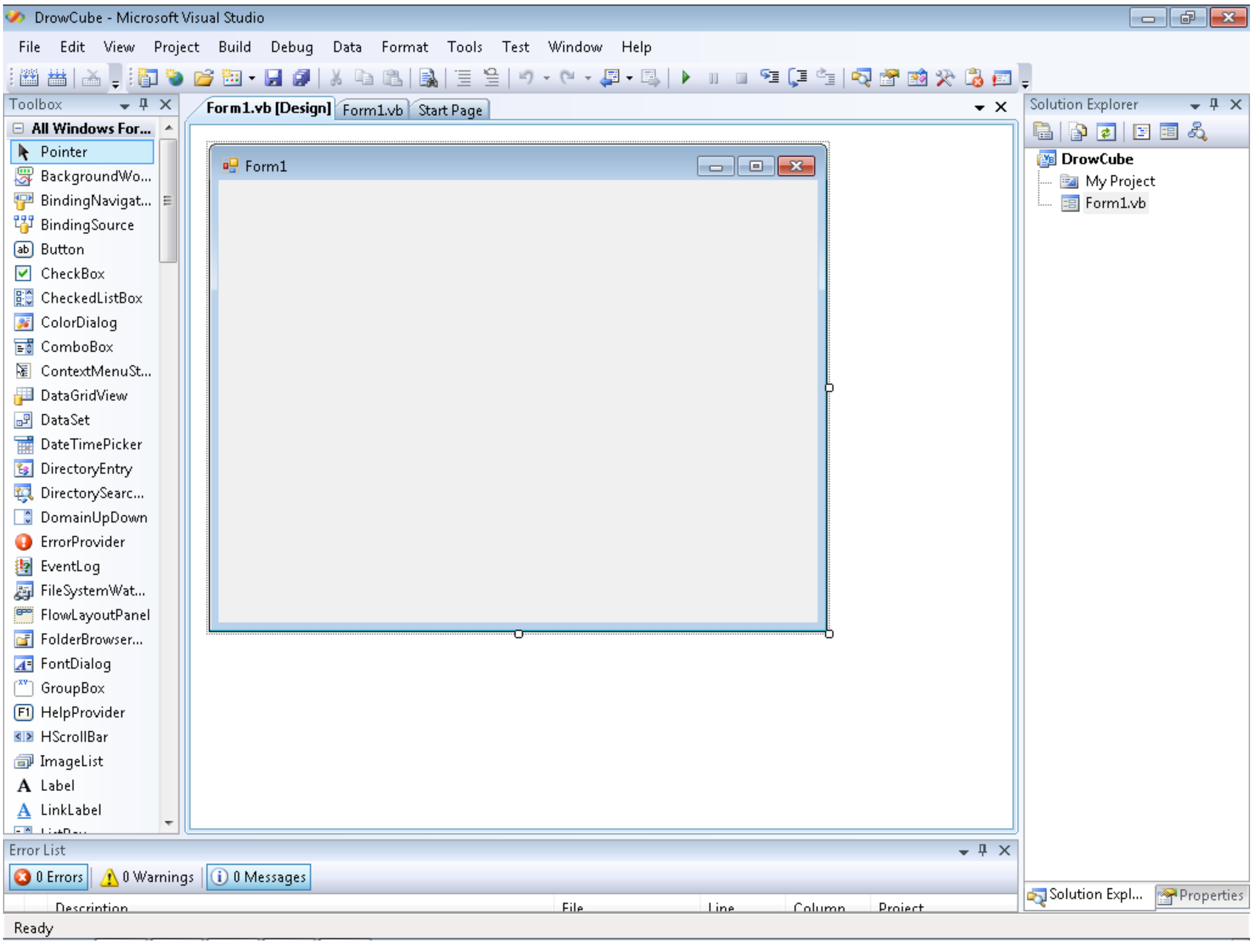
بسم الله و الصلاة و السلام على رسول الله صلى الله عليه و سلم

كيف ترسم مكعب متحرك على نموذج بدون استخدام مكتبة OpenGl

قم بفتح الفيجوال ستديو

انشىء مشروع جديد نوعه Windows Form



قم بالنقر المزدوج على النموذج ليظهر لك صفحة الكود

قم بعد ذلك بمسح جميع ما فى صفحة الكود ثم انسخ الكود التالى مباشرة بدون اى تعديل

```
Option Explicit On

' ترجمان عربى
Imports System.Drawing.Graphics
Imports System.Drawing.Pen
Imports System.Drawing.Color
Imports System.Drawing.Brush
```

```vb
Imports System.Drawing.Point
Imports System.Drawing.Bitmap
Public Class Form1
    Protected m_timer As Timer
    Protected m_vertices(8) As Point3D
    Protected m_faces(6, 4) As Integer
    Protected m_colors(6) As Color
    Protected m_brushes(6) As Brush
    Protected m_angle As Integer

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        ' Enable double-buffering to eliminate flickering.
        Me.SetStyle(ControlStyles.AllPaintingInWmPaint, True)
        Me.SetStyle(ControlStyles.OptimizedDoubleBuffer, True)

        InitCube()
        'انشاء التايمر
        m_timer = New Timer()

        ' ضبط التايمر
        m_timer.Interval = 25

        ' انشاء حدث الذى سنفذه التايمر
        AddHandler m_timer.Tick, AddressOf AnimationLoop

        ' بدء عمل التايمر
        m_timer.Start()
    End Sub
    Private Sub InitCube()
        ' Create the cube vertices.
        m_vertices = New Point3D() { _
                     New Point3D(-1, 1, -1), _
                     New Point3D(1, 1, -1), _
                     New Point3D(1, -1, -1), _
                     New Point3D(-1, -1, -1), _
                     New Point3D(-1, 1, 1), _
                     New Point3D(1, 1, 1), _
                     New Point3D(1, -1, 1), _
                     New Point3D(-1, -1, 1)}

        ' Create an array representing the 6 faces of a cube. Each face is
composed by indices to the vertex array
        ' above.
        m_faces = New Integer(,) {{0, 1, 2, 3}, {1, 5, 6, 2}, {5, 4, 7, 6},
{4, 0, 3, 7}, {0, 4, 5, 1}, {3, 2, 6, 7}}

        ' Define the colors of each face.
        m_colors = New Color() {Color.BlueViolet, Color.Cyan, Color.Green,
Color.Yellow, Color.Violet, Color.LightSkyBlue}

        ' Create the brushes to draw each face. Brushes are used to draw
filled polygons.
        For i = 0 To 5
            m_brushes(i) = New SolidBrush(m_colors(i))
        Next
    End Sub

    Private Sub AnimationLoop()
        ' Forces the Paint event to be called.
        Me.Invalidate()
```

```
        ' Update the variable after each frame.
        m_angle += 1
    End Sub

    Private Sub Main_Paint(ByVal sender As Object, ByVal e As
System.Windows.Forms.PaintEventArgs) Handles Me.Paint
        Dim t(8) As Point3D
        Dim f(4) As Integer
        Dim v As Point3D
        Dim avgZ(6) As Double
        Dim order(6) As Integer
        Dim tmp As Double
        Dim iMax As Integer

        ' Clear the window
        e.Graphics.Clear(Color.LightBlue)

        ' Transform all the points and store them on the "t" array.
        For i = 0 To 7
            Dim b As Brush = New SolidBrush(Color.White)
            v = m_vertices(i)
            t(i) = v.RotateX(m_angle).RotateY(m_angle).RotateZ(Me.m_angle)
            t(i) = t(i).Project(Me.ClientSize.Width, Me.ClientSize.Height,
256, 4)
        Next

        ' Compute the average Z value of each face.
        For i = 0 To 5
            avgZ(i) = (t(m_faces(i, 0)).Z + t(m_faces(i, 1)).Z + t(m_faces(i,
2)).Z + t(m_faces(i, 3)).Z) / 4.0
            order(i) = i
        Next

        ' Next we sort the faces in descending order based on the Z value.
        ' The objective is to draw distant faces first. This is called
        ' the PAINTERS ALGORITHM. So, the visible faces will hide the
invisible ones.
        ' The sorting algorithm used is the SELECTION SORT.
        For i = 0 To 4
            iMax = i
            For j = i + 1 To 5
                If avgZ(j) > avgZ(iMax) Then
                    iMax = j
                End If
            Next
            If iMax <> i Then
                tmp = avgZ(i)
                avgZ(i) = avgZ(iMax)
                avgZ(iMax) = tmp

                tmp = order(i)
                order(i) = order(iMax)
                order(iMax) = tmp
            End If
        Next

        ' Draw the faces using the PAINTERS ALGORITHM (distant faces first,
closer faces last).
        For i = 0 To 5
            Dim points() As Point
            Dim index As Integer = order(i)
```

```
            points = New Point() {New Point(CInt(t(m_faces(index, 0)).X),
CInt(t(m_faces(index, 0)).Y)), _
                New Point(CInt(t(m_faces(index, 1)).X), CInt(t(m_faces(index,
1)).Y)), _
                New Point(CInt(t(m_faces(index, 2)).X), CInt(t(m_faces(index,
2)).Y)), _
                New Point(CInt(t(m_faces(index, 3)).X), CInt(t(m_faces(index,
3)).Y)) _
            }
            e.Graphics.FillPolygon(m_brushes(index), points)
        Next
    End Sub
End Class
Public Class Point3D
    Protected m_x As Double, m_y As Double, m_z As Double

    Public Sub New(ByVal x As Double, ByVal y As Double, ByVal z As Double)
        Me.X = x
        Me.Y = y
        Me.Z = z
    End Sub

    Public Property X() As Double
        Get
            Return m_x
        End Get
        Set(ByVal value As Double)
            m_x = value
        End Set
    End Property

    Public Property Y() As Double
        Get
            Return m_y
        End Get
        Set(ByVal value As Double)
            m_y = value
        End Set
    End Property

    Public Property Z() As Double
        Get
            Return m_z
        End Get
        Set(ByVal value As Double)
            m_z = value
        End Set
    End Property

    Public Function RotateX(ByVal angle As Integer) As Point3D
        Dim rad As Double, cosa As Double, sina As Double, yn As Double, zn
As Double

        rad = angle * Math.PI / 180
        cosa = Math.Cos(rad)
        sina = Math.Sin(rad)
        yn = Me.Y * cosa - Me.Z * sina
        zn = Me.Y * sina + Me.Z * cosa
        Return New Point3D(Me.X, yn, zn)
    End Function

    Public Function RotateY(ByVal angle As Integer) As Point3D
```

```
        Dim rad As Double, cosa As Double, sina As Double, Xn As Double, Zn
As Double

        rad = angle * Math.PI / 180
        cosa = Math.Cos(rad)
        sina = Math.Sin(rad)
        Zn = Me.Z * cosa - Me.X * sina
        Xn = Me.Z * sina + Me.X * cosa

        Return New Point3D(Xn, Me.Y, Zn)
    End Function

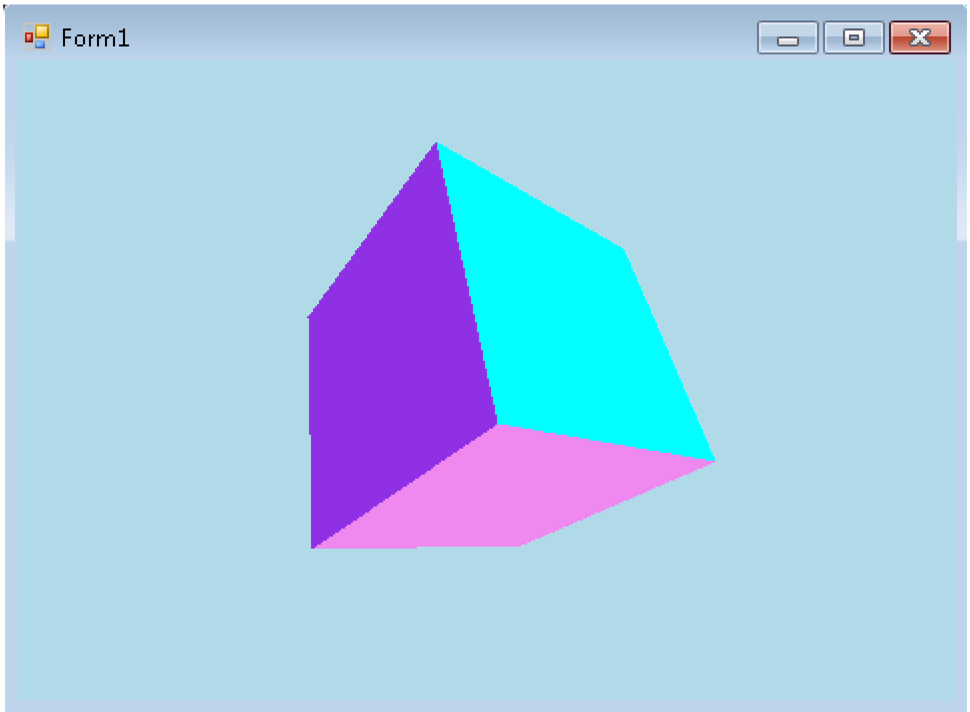
    Public Function RotateZ(ByVal angle As Integer) As Point3D
        Dim rad As Double, cosa As Double, sina As Double, Xn As Double, Yn
As Double

        rad = angle * Math.PI / 180
        cosa = Math.Cos(rad)
        sina = Math.Sin(rad)
        Xn = Me.X * cosa - Me.Y * sina
        Yn = Me.X * sina + Me.Y * cosa
        Return New Point3D(Xn, Yn, Me.Z)
    End Function

    Public Function Project(ByVal viewWidth, ByVal viewHeight, ByVal fov,
ByVal viewDistance)
        Dim factor As Double, Xn As Double, Yn As Double
        factor = fov / (viewDistance + Me.Z)
        Xn = Me.X * factor + viewWidth / 2
        Yn = Me.Y * factor + viewHeight / 2
        Return New Point3D(Xn, Yn, Me.Z)
    End Function
End Class
```

ملحوظة مهمة يجب ان يكون اسم النموذج هو : Form1
و اذا لم يكن كذلك قم بتغيير اسمه الى Form1

اضغط بعد ذلك على F5



محمد احمد عبد الغنى ابراهيم(ترجمان عربى)

Mo.am86@yahoo.com

https://www.facebook.com/mohuomha.mo

تحميل السورس كود الخاص بالمثال

ستجد السورس ضمن مجموعة برامج اخرى على الرابط التالى

http://arabicturgeman.blogspot.com/